DETAILED ACTION

EXAMINER'S AMENDMENT

An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with attorney Robert Kowert on 08/17/2009, followed by applicant's submitted claim draft on the same date.

The claims have been amended as follows:

1. (Currently amended) A system, comprising:

a client comprising a client Web services stack that supports both a markup language protocol and a binary encoding protocol, wherein the markup language protocol is based on XML (eXtensible Markup Language); and

a server comprising a server Web services stack that supports both the markup language protocol and the binary encoding protocol, wherein the server Web services stack is configured to:

communicate with the client Web services stack according to the markup language protocol; and

dynamically switch to communicate with the client Web services stack according to the binary encoding protocol, wherein communication

according to the binary encoding protocol comprises mapping from an XML schema to a binary encoding schema and generating a binary encoding from the binary encoding schema;

wherein the client Web services stack and the server Web services stack each support the markup language protocol and the binary encoding protocol with a single API (application programming interface).

2. (Previously presented) The system as recited in claim 1, wherein the client is a JAX-RPC (Java API for XML (eXtensible Markup Language)-based RPC (Remote Procedure Call)) client.

3. (Previously presented) The system as recited in claim 1, wherein the client is a J2ME (Java 2 Micro Edition) client.

4. (Previously presented) The system as recited in claim 1, wherein the server is a JAX-RPC (Java API for XML (eXtensible Markup Language)-based RPC (Remote Procedure Call)) server.

5. (Canceled)

6. (Currently amended) The system as recited in claim 1, wherein ~~communication according to the binary encoding protocol comprises mapping from a markup language~~

~~schema to~~ the binary encoding schema is an ASN.1 (Abstract Syntax Notation One) schema~~, and generating a binary encoding from the ASN.1 schema~~.

7. (Original) The system as recited in claim 1, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

8. (Original) The system as recited in claim 1, further comprising another client comprising another client Web services stack that supports only the binary encoding protocol, and wherein the server Web services stack is further configured to communicate with the other client Web services stack according to the binary encoding protocol.

9. (Original) The system as recited in claim 1, wherein, to communicate with the client Web services stack according to the binary encoding protocol, the server Web services stack is further configured to:

translate the markup language protocol to binary encoding protocol messages for transmission to the client Web services stack; and

translate binary encoding protocol messages received from the client Web services stack to the markup language protocol.

10. (Previously presented) The system as recited in claim 1, wherein, to communicate with the client Web services stack according to the binary encoding protocol, the server

Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to a self-describing binary format that preserves the markup language protocol information set.

11. (Previously presented) The system as recited in claim 1, wherein, to communicate with the client Web services stack according to the binary encoding protocol, the server Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to a schema-optimized binary format for transmitting data described by markup language schema.

12. (Currently amended) A system, comprising:

a processor; and

a memory comprising program instructions, wherein the program instructions are executable by the processor to implement a Web services stack configured to:

communicate with another Web services stack on another system according to [[the]] a markup language protocol, wherein the markup language protocol is based on XML (eXtensible Markup Language); and

dynamically switch to communicate with the other Web services stack according to the binary encoding protocol, wherein communication according to the binary encoding protocol comprises mapping from

an XML schema to a binary encoding schema and generating a binary encoding from the binary encoding schema;

wherein the Web services stack supports the markup language protocol and the binary encoding protocol with a single API (application programming interface).

13. (Previously presented) The system as recited in claim 12, wherein the system is a JAX-RPC (Java API for XML (eXtensible Markup Language)-based RPC (Remote Procedure Call)) client.

14. (Previously presented) The system as recited in claim 12, wherein the system is a J2ME (Java 2 Micro Edition) client.

15. (Previously presented) The system as recited in claim 12, wherein the system is a JAX-RPC (Java API for XML (eXtensible Markup Language)-based RPC (Remote Procedure Call)) server.

16. (Original) The system as recited in claim 12, wherein the system and the other system are peers on a network.

17. (Canceled)

18. (Currently amended) The system as recited in claim 12, wherein ~~communication~~

~~according to the binary encoding protocol comprises mapping from a markup language schema to~~ the binary encoding schema is an ASN.1 (Abstract Syntax Notation One) schema~~, and generating a binary encoding from the ASN.1 schema~~.

19. (Original) The system as recited in claim 12, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

20. (Original) The system as recited in claim 12, wherein, to communicate with the other Web services stack according to the binary encoding protocol, the Web services stack is further configured to:

translate the markup language protocol to binary encoding protocol messages for transmission to the other Web services stack; and

translate binary encoding protocol messages received from the other Web services stack to the markup language protocol.

21. (Previously presented) The system as recited in claim 12, wherein, to communicate with the other Web services stack according to the binary encoding protocol, the Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to a self-describing binary format that preserves the markup language protocol information set.

22. (Previously presented) The system as recited in claim 12, wherein, to communicate

with the other Web services stack according to the binary encoding protocol, the Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to a schema-optimized binary format for transmitting data described by markup language schema.

23. (Currently amended) A system, comprising:

a client system, server system, or peer device comprising:

means for communicating between a Web services stack on the system and another Web services stack on another system according to a markup language protocol, wherein the markup language protocol is based on XML (eXtensible Markup Language); and

means for dynamically switching to communicate between the Web services stack and the other Web services stack according to a binary encoding protocol, wherein communication according to the binary encoding protocol comprises mapping from an XML schema to a binary encoding schema and generating a binary encoding from the binary encoding schema;

wherein the Web services stack supports the markup language protocol and the binary encoding protocol with a single API (application programming interface).

24. (Currently amended) The system as recited in claim 23, wherein ~~the markup~~

~~language protocol is XML (eXtensible Markup Language), and wherein communication according to the binary encoding protocol comprises mapping from an XML (eXtensible Markup Language) schema to~~ the binary encoding schema is an ASN.1 (Abstract Syntax Notation One) schema~~, and generating a binary encoding from the ASN.1 schema~~.

25. (Original) The system as recited in claim 24, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

26. (Currently amended) A method, comprising:

a Web services stack communicating with another Web services stack according to a markup language protocol, wherein the markup language protocol is based on XML (eXtensible Markup Language); and

the Web services stack dynamically switching to communicating with the other Web services stack according to a binary encoding protocol, wherein communication according to the binary encoding protocol comprises mapping from an XML schema to a binary encoding schema and generating a binary encoding from the binary encoding schema;

wherein the Web services stack supports the markup language protocol and the binary encoding protocol with a single API (application programming interface).

27. (Previously presented) The method as recited in claim 26, wherein the Web services stack is implemented on a JAX-RPC (Java API for XML (eXtensible Markup Language)-based RPC (Remote Procedure Call)) client system.

28. (Previously presented) The method as recited in claim 26, wherein the Web services stack is implemented on a J2ME (Java 2 Micro Edition) client system.

29. (Previously presented) The method as recited in claim 26, wherein the Web services stack is implemented on a JAX-RPC (Java API for XML (eXtensible Markup Language)-based RPC (Remote Procedure Call)) server system.

30. (Original) The method as recited in claim 26, wherein the Web services stack and the other Web services stack are implemented on peers on a network.

31. (Canceled)

32. (Currently amended) The method as recited in claim 26, wherein ~~communicating according to the binary encoding protocol comprises mapping from a markup language schema to~~ the binary encoding schema is an ASN.1 (Abstract Syntax Notation One) schema~~, and generating a binary encoding from the ASN.1 schema~~.

33. (Original) The method as recited in claim 26, wherein the binary encoding protocol

uses Packed Encoding Rules (PER) encoding.

34. (Original) The method as recited in claim 26, wherein said communicating with the other Web services stack according to the binary encoding protocol comprises:

translating the markup language protocol to binary encoding protocol messages for transmission to the other Web services stack; and

translating binary encoding protocol messages received from the other Web services stack to the markup language protocol.

35. (Previously presented) The method as recited in claim 26, wherein said communicating with the other Web services stack according to the binary encoding protocol comprises serializing the markup language protocol to generate binary encoding protocol messages according to a self-describing binary format that preserves the markup language protocol information set.

36. (Previously presented) The method as recited in claim 26, wherein said communicating with the other Web services stack according to the binary encoding protocol comprises serializing the markup language protocol to generate binary encoding protocol messages according to a schema-optimized binary format for transmitting data described by markup language schema.

37. (Currently amended) A computer-accessible storage medium comprising program

instructions, wherein the program instructions are configured to implement:

a Web services stack communicating with another Web services stack according to a markup language protocol, wherein the markup language protocol is based on XML (eXtensible Markup Language); and

the Web services stack dynamically switching to communicating with the other Web services stack according to a binary encoding protocol, wherein communication according to the binary encoding protocol comprises mapping from an XML schema to a binary encoding schema and generating a binary encoding from the binary encoding schema;

wherein the Web services stack supports the markup language protocol and the binary encoding protocol with a single API (application programming interface).

38. (Previously presented) The computer-accessible storage medium as recited in claim 37, wherein the Web services stack is implemented on a JAX-RPC (Java API for XML (eXtensible Markup Language)-based RPC (Remote Procedure Call)) client system.

39. (Previously presented) The computer-accessible storage medium as recited in claim 37, wherein the Web services stack is implemented on a J2ME (Java 2 Micro Edition) client system.

40. (Previously presented) The computer-accessible storage medium as recited in claim

37, wherein the Web services stack is implemented on a JAX-RPC (Java API for XML (eXtensible Markup Language)-based RPC (Remote Procedure Call)) server system.

41. (Previously presented) The computer-accessible storage medium as recited in claim 37, wherein the Web services stack and the other Web services stack are implemented on peers on a network.

42. (Canceled)

43. (Currently amended) The computer-accessible storage medium as recited in claim 37, wherein ~~communicating according to the binary encoding protocol comprises mapping from a markup language schema to~~ the binary encoding schema is an ASN.1 (Abstract Syntax Notation One) schema~~, and generating a binary encoding from the ASN.1 schema~~.

44. (Previously presented) The computer-accessible storage medium as recited in claim 37, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

45. (Previously presented) The computer-accessible storage medium as recited in claim 37, wherein, in said communicating with the other Web services stack according to the binary encoding protocol, the program instructions are further configured to implement:

translating the markup language protocol to binary encoding protocol messages

for transmission to the other Web services stack; and

translating binary encoding protocol messages received from the other Web services stack to the markup language protocol.

46. (Previously presented) The computer-accessible storage medium as recited in claim 37, wherein, in said communicating with the other Web services stack according to the binary encoding protocol, the program instructions are further configured to implement serializing the markup language protocol to generate binary encoding protocol messages according to a self-describing binary format that preserves the markup language protocol information set.

47. (Previously presented) The computer-accessible storage medium as recited in claim 37, wherein, in said communicating with the other Web services stack according to the binary encoding protocol, the program instructions are further configured to implement serializing the markup language protocol to generate binary encoding protocol messages according to a schema-optimized binary format for transmitting data described by markup language schema.

48. (Currently amended) A system, comprising:

a processor; and

a memory comprising program instructions, wherein the program instructions are executable by the processor to implement a Web services stack

configured to:

communicate with other systems using either a binary encoding protocol or a markup language protocol using a single API (application programming interface), wherein the markup language protocol is based on XML (eXtensible Markup Language), and wherein communication according to the binary encoding protocol comprises mapping from an XML schema to a binary encoding schema and generating a binary encoding from the binary encoding schema;

negotiate with another system to determine if the other system supports the binary encoding protocol;

if the other system supports the binary encoding protocol, communicate with the other system according to the binary encoding protocol; and

if the other system does not support the binary encoding protocol, communicate with the other system according to the markup language protocol.

49. (Previously presented) The system as recited in claim 48, wherein the system is a JAX-RPC (Java API for XML (eXtensible Markup Language)-based RPC (Remote Procedure Call)) server.

50. (Original) The system as recited in claim 48, wherein the system and the other system are peers on a network.

51. (Canceled)

52. (Currently amended) The system as recited in claim 48, wherein ~~communication according to the binary encoding protocol comprises mapping from a markup language schema to~~ the binary encoding schema is an ASN.1 (Abstract Syntax Notation One) schema~~, and generating a binary encoding from the ASN.1 schema~~.

53. (Original) The system as recited in claim 48, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

54. (Original) The system as recited in claim 48, wherein the Web services stack is further configured to, if the other system includes a Web services stack configured to communicate with either the binary encoding protocol or the markup language protocol:

communicate with the other system according to the markup language protocol; and

dynamically switch to communicate with the other system according to the binary encoding protocol.

55. (Previously presented) The system as recited in claim 48, wherein, to communicate

with the other system according to the binary encoding protocol, the Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to a self-describing binary format that preserves the markup language protocol information set.

56. (Previously presented) The system as recited in claim 48, wherein, to communicate with the other system according to the binary encoding protocol, the Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to a schema-optimized binary format for transmitting data described by markup language schema.

57. (Currently amended) A system, comprising:

a client system, server system, or peer device comprising:

means for communicating with other systems using either a binary encoding protocol or a markup language protocol using a single API (application programming interface), wherein the markup language protocol is based on XML (eXtensible Markup Language), and wherein communication according to the binary encoding protocol comprises mapping from an XML schema to a binary encoding schema and generating a binary encoding from the binary encoding schema;

means for negotiating with another system to determine if the other system supports the binary encoding protocol; and

means for communicating with the other system according to the binary encoding protocol if the other system supports the binary encoding protocol.

58. (Currently amended) The system as recited in claim 57, wherein ~~the markup language protocol is XML (eXtensible Markup Language), and wherein communication according to the binary encoding protocol comprises mapping from an XML (eXtensible Markup Language) schema to~~ the binary encoding schema is an ASN.1 (Abstract Syntax Notation One) schema~~, and generating a binary encoding from the ASN.1 schema~~.

59. (Original) The system as recited in claim 57, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

60. (Currently amended) A method, comprising:

a Web services stack on a system configured to communicate with other systems using either a binary encoding protocol or a markup language protocol using a single API (application programming interface) negotiating with another system to determine if the other system supports the binary encoding protocol, wherein the markup language protocol is based on XML (eXtensible Markup Language), and wherein communication according to the binary encoding protocol comprises mapping from an XML schema to a binary encoding schema and generating a binary

encoding from the binary encoding schema;

if the other system supports the binary encoding protocol, the Web services stack communicating with the other system according to the binary encoding protocol; and

if the other system does not support the binary encoding protocol, the Web services stack communicating with the other system according to the markup language protocol.

61. (Previously presented) The method as recited in claim 60, wherein the system is a JAX-RPC (Java API for XML (eXtensible Markup Language)-based RPC (Remote Procedure Call)) server.

62. (Original) The method as recited in claim 60, wherein the system and the other system are peers on a network.

63. (Currently amended) The method as recited in claim 60, wherein ~~the markup language protocol is XML (eXtensible Markup Language), and wherein communicating according to the binary encoding protocol comprises mapping from an XML (eXtensible Markup Language) schema to~~ the binary encoding schema is an ASN.1 (Abstract Syntax Notation One) schema~~, and generating a binary encoding from the ASN.1 schema~~.

64. (Original) The method as recited in claim 60, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

65. (Original) The method as recited in claim 60, further comprising, if the other system includes a Web services stack configured to communicate with either the binary encoding protocol or the markup language protocol:

communicating with the other system according to the markup language protocol; and

dynamically switching to communicating with the other system according to the binary encoding protocol.

66. (Previously presented) The method as recited in claim 60, wherein said communicating with the other system according to the binary encoding protocol comprises serializing the markup language protocol to generate binary encoding protocol messages according to a self-describing binary format that preserves the markup language protocol information set.

67. (Previously presented) The method as recited in claim 60, wherein said communicating with the other system according to the binary encoding protocol comprises serializing the markup language protocol to generate binary encoding protocol messages according to a schema-optimized binary format for transmitting data described by markup language schema.

68. (Currently amended) A computer-accessible storage medium comprising program instructions, wherein the program instructions are configured to implement:

a Web services stack on a system configured to communicate with other systems using either a binary encoding protocol or a markup language protocol using a single API (application programming interface) negotiating with another system to determine if the other system supports the binary encoding protocol, wherein the markup language protocol is based on XML (eXtensible Markup Language), and wherein communication according to the binary encoding protocol comprises mapping from an XML schema to a binary encoding schema and generating a binary encoding from the binary encoding schema;

if the other system supports the binary encoding protocol, the Web services stack communicating with the other system according to the binary encoding protocol; and

if the other system does not support the binary encoding protocol, the Web services stack communicating with the other system according to the markup language protocol.

69. (Previously presented) The computer-accessible storage medium as recited in claim 68, wherein the system is a JAX-RPC (Java API for XML (eXtensible Markup Language)-based RPC (Remote Procedure Call)) server.

70. (Previously presented) The computer-accessible storage medium as recited in claim 68, wherein the system and the other system are peers on a network.

71. (Currently amended) The computer-accessible storage medium as recited in claim 68, wherein ~~the markup language protocol is XML (eXtensible Markup Language), and wherein communicating according to the binary encoding protocol comprises mapping from an XML (eXtensible Markup Language) schema to~~ the binary encoding schema is an ASN.1 (Abstract Syntax Notation One) schema~~, and generating a binary encoding from the ASN.1 schema~~.

72. (Previously presented) The computer-accessible storage medium as recited in claim 68, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

73. (Previously presented) The computer-accessible storage medium as recited in claim 68, wherein the program instructions are further configured to implement, if the other system includes a Web services stack configured to communicate with either the binary encoding protocol or the markup language protocol:

communicating with the other system according to the markup language protocol; and

dynamically switching to communicating with the other system according to the binary encoding protocol.

74. (Previously presented) The computer-accessible storage medium as recited in claim 68, wherein, in said communicating with the other system according to the binary encoding protocol, the program instructions are further configured to implement serializing the markup language protocol to generate binary encoding protocol messages according to a self-describing binary format that preserves the markup language protocol information set.

75. (Previously presented) The computer-accessible storage medium as recited in claim 68, wherein, in said communicating with the other system according to the binary encoding protocol, the program instructions are further configured to implement serializing the markup language protocol to generate binary encoding protocol messages according to a schema-optimized binary format for transmitting data described by markup language schema.

Reasons for Allowance

The following is an examiner's statement of reasons for allowance:

Claims 1-4, 6-16, 18-30, 32-41, 43-50, 52-75 are allowed. The prior art of record does not teach the claimed invention, as follows.

For all independent claims, the prior art does not teach a method/system for negotiating a XML-based protocol or a binary encoding protocol between two web

services stacks; and if the binary encoding protocol is supported, a method/system switches the Web services stack communicating with the other system according to the binary encoding protocol which is preferable to the XML-based protocol; wherein one single API in the client/server supports both XML and the binary encoding protocol. The prior art of record discloses both XML and a binary encoding protocol as possible communication protocols. However, the prior art of record fails to disclose or suggest negotiation of both XML and the binary protocol support capability in both client and server web services stacks in order to dynamically switch to the binary encoding protocol.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure is included in form PTO 892.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Hieu T. Hoang whose telephone number is 571-270-

1253. The examiner can normally be reached on Monday-Thursday, 8 a.m.-5 p.m., EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Follansbee can be reached on 571-272-3964. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

HH

/Kenny S Lin/

Primary Examiner, Art Unit 2452